Pivotal™

Rev: A01
Updated: May 15, 2014

Welcome to Greenplum Database 4.2.7.3

Greenplum Database is a massively parallel processing (MPP) database server that supports next generation data warehousing and large-scale analytics processing. By automatically partitioning data and running parallel queries, it allows a cluster of servers to operate as a single database supercomputer performing tens or hundreds times faster than a traditional database. It supports SQL, MapReduce parallel processing, and data volumes ranging from hundreds of gigabytes, to hundreds of terabytes.

Note: This document contains pertinent release information about Greenplum Database 4.2.7.3. For previous versions of the release notes for Greenplum Database, go to Support Zone.

About Pivotal, Inc.

Greenplum is currently transitioning to a new corporate identity (Pivotal Software, Inc.). During this transition, there will be some legacy instances of our former corporate identity (Greenplum) appearing in our products and documentation. If you have any questions or concerns, please do not hesitate to contact us through our web site:
http://www.gopivotal.com/contact

About Greenplum Database 4.2.7.3

Greenplum Database 4.2.7.3 is a maintenance that contains a critical fix. Please refer to the following sections for more information about this release.

- Downloading Greenplum Database
- Changed Feature
- Supported Platforms
- Resolved Issues in Greenplum Database 4.2.7.3
- Known Issues in Greenplum Database 4.2.7.3
- Upgrading to Greenplum Database 4.2.x.x
- Greenplum Database Tools Compatibility
- Greenplum Database Extensions Compatibility
- Hadoop Distribution Compatibility
- Greenplum Database 4.2 Documentation

Downloading Greenplum Database

The Greenplum Database software and documentation can be accessed from the following locations:

- Greenplum Database 4.2.7.3 server software, and other Greenplum Database 4.2.7.3 software including tools and extension packages, can be downloaded from the EMC Download Center.
- Documentation for Greenplum Database 4.2.7.3 is available from the Pivotal Documentation site.

  Documentation for previous release versions of Greenplum Database, as well as other Greenplum Database documentation, is available from Support Zone.

Changed Feature

Greenplum Database 4.2.7.3 supports:

- Red Hat Enterprise Linux 6.5 64-bit.
- SUSE 11 SP2

See "Supported Platforms" for information about operating system support.

Supported Platforms

Greenplum Database 4.2.7.3 runs on the following platforms:

- Red Hat Enterprise Linux 64-bit 5.5, 5.6, 5.7, 6.1, 6.2, 6.4, and 6.5
- SUSE Linux Enterprise Server 64-bit 10 SP4, 11 SP1, and 11 SP2
- Solaris x86 64-bit v10 U7, U8, U9, U10
- Oracle Unbreakable Linux 64-bit 5.5
- CentOS 64-bit 5.5, 5.6, 5.7, 6.1, and 6.2

Greenplum Database 4.2.7.3 supports Data Domain Boost on Red Hat Enterprise Linux.

Greenplum Database 4.2.7.3 supports Data Domain Boost SDK version 2.6.2.0 with DDOS 5.2, 5.3 and 5.4.

GPDB 4.2.7.3 was certified against the following DCA versions:

- DCA v1 - ISO_VERSION="1.2.1.1"
- DCA v2 - ISO_VERSION="2.0.2.0"

Note: For the most recent server platform support information for Greenplum Database, check the EMC Support Matrix (ESM).

To access the ESM, go to the Support Zone home page and click the link **E-Lab Interoperability Navigator**.

In the E-Lab Interoperability Navigator, search for Greenplum Database and add one or more search results to the search cart. Click **Get Results** to display links to EMC Support Statements.

Resolved Issues in Greenplum Database 4.2.7.3

The table below lists issues that are now resolved in Greenplum Database 4.2.7.3. For issues resolved in prior releases, refer to the corresponding release notes available from Support Zone.

Table 1 Resolved Issues in 4.2.7.3

Issue Number	Category	Resolved in	Description
22787	Monitoring: gpperfmon server	4.2.7.3	In some cases, the Greenplum Database gpmmon process failed. The gpmmon process is used for Greenplum Database performance monitoring.
22572	Loaders: Copy/External Tables	4.2.7.2	In some cases, SQL queries intermittently failed when reading from external web tables and the web table was serving data over HTTP from a traditional web server.
22442	Loaders: gpfdist	4.2.7.2	The Greenplum Database Load Tools for Windows installation did not include the gssapi and auth libraries. This issue has been resolved.
22317	Loaders: gpfdist	4.2.7.2	When the Greenplum Database gpfdist utility was used with a named pipe for a readable external table, and there was no data from the pipe for more than 10 minutes, gpfdist failed.
22293	Backup and Restore	4.2.7.2	Greenplum Database supports Data Domain DDOS 5.4. See Supported Platforms for information about supported versions of Data Domain Boost.
22219 20076	Query Planner	4.2.7.2	In certain queries that contain the median function and a GROUP BY clause, the query planner produced an incorrect plan in which some necessary columns were not projected in the operator nodes. This caused an error when trying to look up the missing columns.
22266	Parser	4.2.7.1	When processing an SQL query, it was incorrectly assumed that the data type for DISTINCT columns in a subquery took the outer query data type when the outer query was part of a UNION. These types of queries might have returned an error.
22254	Monitoring: gpperfmon server	4.2.7.1	In some cases, Greenplum Command Center displayed incorrect CPU system information in the System Metrics tab.
22210	Backup and Restore	4.2.7.1	When restoring sets of tables with the Greenplum Database gpdbrestore utility, primary keys, indexes, and GRANT privileges were not restored for filtered tables. For tables with non-English characters (for example, table name and records contained Korean or Chinese characters) primary keys, indexes and GRANT privileges were not restored. For tables with English characters, GRANT privileges were not restored.
22077	Transaction Management	4.2.7.1	A Greenplum Database internal function caused a buffer overrun when a combination of the following information created a very long string: user name, database name, and client host information.
22072	Backup and Restore	4.2.7.1	An incremental backup created with gpcrondump did not generate the correct restore list if some tables were dropped after an ACCESS SHARED lock was released. Moved the lock on pg_class to the gpcrondump utility from the gp_dump utility for better consistency.

Table 1 Resolved Issues in 4.2.7.3

Issue Number	Category	Resolved in	Description
22060	Functions and Languages, Resource Management	4.2.7.1	The string_to_array function fails to produce an array when the input string is larger than 250MB.
22045	Dispatch	4.2.7.1	Reporting errors when the maximum number of connections on Greenplum Database segment instances were exceeded has been enhanced. • Error message for failed connections to segment instances are printed to pg_log without enabling debug mode. • The error message displayed on the client is segworker group creation failed.
21880	Dispatch	4.2.7.1	An internal error message was displayed for an error that occurred for an unsupported query plan with multiple writer segworker groups. The message has been enhanced and provides a description and more information about the context. This is the updated error message. ERROR: Unsupported query plan with multiple writer gangs. HINT: Likely caused by a user defined function that reads or modifies data in a distributed table. CONTEXT: SQL statement "*SQL_statement*"
21795	Fault Detection and Transitions	4.2.7.1	The Greenplum Database server configuration parameter gp_log_fts specifies the level of fault probing messages written to the log file. Valid values are off, terse, verbose, and debug. The verbose mode provides minimal useful log messages. After the value of gp_log_fts has been changed, stop and restart Greenplum Database with the command gpstop -u for the value to take affect.
21694	Query Execution	4.2.7.1	When a Hash Join or Hash Aggregate operator spilled to disk for an SQL query, the amount of workfile bytes read and written to disk was incorrectly reported in the EXPLAIN ANALYZE output.
21630 19106	Resource Management	4.2.7.1	Dropping the resource queue used by an active session from another session would result in a postmaster reset on the master. This has been changed. Now, instead of a reset, an error message is generated in the affected session.
21545	Access Methods	4.2.7.1	On some occasions when a pg_terminate_backend command was issued, Greenplum Database encounters a PANIC and performs a recovery.
21050	Monitoring: gpperfmon server, Monitoring: gpperfmon UI	4.2.7.1	Under some conditions, the Greenplum Command Center process gpsmon would consume a large amount of memory.
20997	Monitoring: gpperfmon server, Monitoring: gpperfmon UI	4.2.7.1	The performance of Greenplum Command Center was slow when processing log alerts from a very large log file. The log processing has been enhanced and the Greenplum Database server configuration parameter gpperfmon_log_alert_level can control the growth of the gpperfmon log.
20970	Query Execution	4.2.7.1	When Greenplum Command Center had been running for a long time, the query it used for disk usage information could not be stopped.

Table 1 Resolved Issues in 4.2.7.3

Issue Number	Category	Resolved in	Description
20466	Catalog and Metadata	4.2.7.1	If a constraint is created with the same name as an existing constraint, the only way to remove the second constraint was to drop all constraints. The fix is to not allow duplicate constraint names. While creating a new table or alter table for new constraint creation, duplicate constraint name are not allowed.
19965	Query Execution	4.2.7.1	In rare cases, wrong results were returned for a query when during query processing the result of an operation is shared between multiple parts of a query through a Shared Scan operator, and the readers of the Shared Scan appear on both sides of a Join operator, and at least two of the readers appear in the same plan slice. In this case, one of the readers might read only zero tuples from the Shared Scan operator. This has been fixed in this release. The fix for this issue might increase the memory consumption for certain queries where these conditions hold.
19930	Monitoring: gpperfmon server	4.2.7.1	In some cases, the Greenplum Command Center web server process could consume a very large amount of memory. This caused the host machine to run out of memory.
18646	Query Execution	4.2.7.1	For some SQL queries that generated a query plan with Shared Scan nodes above a Materialize node, the memory for the Shared Scan node was being freed twice. In some cases, this caused a SEGV.
18635	Monitoring: gpperfmon server	4.2.7.1	In some cases, Greenplum Command Center incorrectly listed SQL queries as active that were no longer active.
21725	Backup and Restore	4.2.7	The following gpcrondump utility issues were caused when the root table of a partitioned table is in a different schema than one of its partitions (child tables) in a database: • Creating a backup with the gpcrondump utility and specifying a list of tables failed. • Creating a backup of a database with the gpcrondump utility succeeded. However, the database could not be restored.
21625	Dispatch	4.2.7	It was possible to send a query cancel error message from a QE process to the QD process for a query that had completed. This caused multiple statement executions to be out of sync between the QE and the QD. In some cases, this led to the QE process to be hung indefinitely.
21522	Backup and Restore	4.2.7	The Greenplum Database utility pg_dump printed information-level messages (messages with the label [INFO]) to stderr that were not printed in previous releases. These messages were printed even when pg_dump completed without errors.
19476	Client Access Methods and Tools	4.2.7	Running multiple gpload sessions simultaneously that loaded data into the same table resulted in inconsistent data in the table.
19259	Transaction Management	4.2.7	During high load situations, the clean up of an out-of-memory condition was not handled correctly and caused a PANIC signal to be issued.
18858	Fault Detection and Transitions	4.2.7	Greenplum Database hung indefinitely due a disk that was failing or had failed. Monitoring for hardware issues has been enhanced in Greenplum Database.

Table 1 Resolved Issues in 4.2.7.3

Issue Number	Category	Resolved in	Description
18623	DDL and Utility Statements	4.2.7	When an out-of-memory condition occurred during a hash table expansion, the hash entry that was being created could have been in an incomplete state. This resulted in misbehavior in the cleanup code.
18576	Transaction Management	4.2.7	When an out-of-memory condition occurred on a lock acquisition code, a hash table entry for the lock record could have been in an incomplete state. This resulted in an invalid pointer handling during clean up.

Known Issues in Greenplum Database 4.2.7.3

This section lists the known issues in Greenplum Database 4.2.7.3. A workaround is provided where applicable.

For known issues discovered in previous releases, including patch releases to Greenplum Database 4.1 or 4.0.x, see the corresponding release notes, available from Support Zone:

Table 2 All Known Issues in 4.2.7.3

Issue	Category	Description
22413	Query Planner	**In some cases, an SQL query that contains the following returns incorrect results: a combination of a median function with other aggregates where the GROUP BY columns are a subset of the table's distribution columns.** **The planner does not correctly redistribute the data to ensure proper collocation of the join tuples, and as a result the join produces incorrect results.**
22560	Catalog and Metadata	**In PL/pgSQL user-defined functions, exceptions are not handled properly. If the function contains exception blocks, the function should not issue DDL commands, TRUNCATE commands, or inserts into append-only column-oriented tables.**
22442	Loaders: gpfdist	**The Greenplum Database Load Tools for Windows is missing gssapi and auth libraries.** **Workaround:** Install the Greenplum Database Client Tools for Windows. The Client Tools include the missing libraries. The Load Tools will use the libraries from the Client Tools.
22172	DDL and Utility Statements	**OLAP queries with a HAVING clause that filters for NULL on dimension columns incorrectly return empty results.**

20453	Query Planner	**For SQL queries of either of the following forms:**
		`SELECT columns FROM table WHERE table.column NOT IN subquery;`
		`SELECT columns FROM table WHERE table.column = ALL subquery;`
		tuples that satisfy both of the following conditions are not included in the result set:
		• *table.column* **is NULL.**
		• *subquery* **returns the empty result.**

21724	Query Planner	**Greenplum Database executes an SQL query in two stages if a scalar subquery is involved. The output of the first stage plan is fed into the second stage plan as a external parameter. If the first stage plan generates zero tuples and directly contributes to the output of the second stage plan, incorrect results might be returned.**
21838	Backup and Restore	**When restoring sets of tables with the Greenplum Database utility gpdbrestore, the table schemas must be defined in the database. If a table's schema is not defined in the database, the table is not restored. When performing a full restore, the database schemas are created when the tables are restored.** **Workaround:** Before restoring a set of tables, create the schemas for the tables in the database.
21129	DDL and Utility Statements	**SSL is only supported on the master host. It is not supported on segment hosts.**

Table 2 All Known Issues in 4.2.7.3

Issue	Category	Description
20822	Backup and Restore	**Special characters such as !, $, #, and @ cannot be used in the password for the Data Domain Boost user when specifying the Data Domain Boost credentials with the `gpcrondump` options `--ddboost-host` and `--ddboost-user`.**
18247	DDL and Utility Statements	**`TRUNCATE` command does not remove rows from a sub-table of a partitioned table. If you specify a sub-table of a partitioned table with the `TRUNCATE` command, the command does not remove rows from the sub-table and its child tables.** **Workaround**: Use the `ALTER TABLE` command with the `TRUNCATE PARTITION` clause to remove rows from the sub-table and its child tables.
19788	Replication: Resync, Transaction Management	**In some rare circumstances, performing a full recovery with `gprecoverseg` fails due to inconsistent LSN.** **Workaround**: Stop and restart the database. Then perform a full recovery with `gprecoverseg`.
19772	Interconnect	**After installing Greenplum Database 4.2.4, downgrading the Greenplum Database installation to a previous minor version causes a crash.** **Workaround**: Before downgrading Greenplum Database 4.2.4 to a previous minor version, change the value for the Greenplum Database parameter `gp_interconnect_type` from `UDPIFC` to a supported value such as `UDP`. The parameter value `UDPIFC` was introduced in Greenplum Database 4.2.4 and is not valid in previous versions.
19705	Loaders: gpload	**`gpload` fails on Windows XP with Python 2.6.** **Workaround**: Install Python 2.5 on the system where gpload is installed.
19493 19464 19426	Backup and Restore	**The `gpcrondump` and `gpdbrestore` utilities do not handle errors returned by DD Boost or Data Domain correctly.** These are two examples: • If invalid Data Domain credentials are specified when setting the Data Domain Boost credentials with the `gpcrondump` utility, the error message does not indicate that invalid credentials were specified. • Restoring a Greenplum database from a Data Domain system with `gpdbrestore` and the `--ddboost` option indicates success even though segment failures occured during the restore. **Workaround**: The errors are logged in the master and segment server backup or restore status and report files. Scan the status and report files to check for error messages.
19278	Backup and Restore	**When performing a selective restore of a partitioned table from a full backup with `gpdbrestore`, the data from leaf partitions are not restored.** **Workaround**: When doing a selective restore from a full backup, specify the individual leaf partitions of the partitioned tables that are being restored. Alternatively, perform a full backup, not a selective backup.
15692 17192	Backup and Restore	**Greenplum Database's implementation of RSA lock box for Data Domain Boost changes backup and restore requirements for customers running SUSE.** The current implementation of the RSA lock box for Data Domain Boost login credential encryption only supports customers running on Red Hat Enterprise Linux. **Workaround**: If you run Greenplum Database on SUSE, use NFS as your backup solution. See the *Greenplum Database System Administrator Guide* for information on setting up a NFS backup.

Table 2 All Known Issues in 4.2.7.3

Issue	Category	Description
18850	Backup and Restore	Data Domain Boost credentials cannot be set up in some environments due to the absence of certain libraries (for example, `libstdc++`) expected to reside on the platform. **Workaround**: Install the missing libraries manually on the system.
18851	Backup and Restore	When performing a data-only restore of a particular table, it is possible to introduce data into Greenplum Database that contradicts the distribution policy of that table. In such cases, subsequent queries may return unexpected and incorrect results. To avoid this scenario, we suggest you carefully consider the table schema when performing a restore.
18774	Loaders	External web tables that use IPv6 addresses must include a port number.
18713	Catalog and Metadata	Drop language plpgsql cascade results in a loss of `gp_toolkit` functionality. **Workaround**: Reinstall `gp_toolkit`.
18710	Management Scripts Suite	Greenplum Management utilities cannot parse IPv6 IP addresses. **Workaround**: Always specify IPv6 hostnames rather than IP addresses
18703	Loaders	The bytenum field (byte offset in the load file where the error occurred) in the error log when using gpfdist with data in text format errors is not populated, making it difficult to find the location of an error in the source file.
12468	Management Scripts Suite	`gpexpand --rollback` fails if an error occurs during expansion such that it leaves the database down `gpstart` also fails as it detects that expansion is in progress and suggests to run `gpexpand --rollback` which will not work because the database is down. **Workaround**: Run `gpstart -m` to start the master and then run rollback,
18785	Loaders	Running `gpload` with the `--ssl` option and the relative path of the source file results in an error that states the source file is missing. **Workaround**: Provide the full path in the yaml file or add the loaded data file to the certificate folder.
18414	Loaders	Unable to define external tables with fixed width format and empty line delimiter when file size is larger than gpfdist chunk (by default, 32K).
14640	Backup and Restore	`gpdbrestore` outputting incorrect non-zero error message. When performing single table restore, `gpdbrestore` gives warning messages about non-zero tables however prints out zero rows.
17285	Backup and Restore	NFS backup with `gpcrondump -c` can fail. In circumstances where you haven't backed up to a local disk before, backups to NFS using `gpcrondump` with the `-c` option can fail. On fresh systems where a backup has not been previously invoked there are no dump files to cleanup and the `-c` flag will have no effect. **Workaround**: Do not run `gpcrondump` with the `-c` option the first time a backup is invoked from a system.
17837	Upgrade/ Downgrade	Major version upgrades internally depend on the `gp_toolkit` system schema. The alteration or absence of this schema may cause upgrades to error out during preliminary checks. **Workaround**: To enable the upgrade process to proceed, you need to reinstall the `gp_toolkit` schema in all affected databases by applying the SQL file found here: `$GPHOME/share/postgresql/gp_toolkit.sql`.

Table 2 All Known Issues in 4.2.7.3

Issue	Category	Description
17513	Management Scripts Suite	Running more than one `gpfilespace` command concurrently with itself to move either temporary files (`--movetempfilespace`) or transaction files (`--movetransfilespace`) to a new filespace can in some circumstances cause OID inconsistencies. **Workaround**: Do not run more than one `gpfilespace` command concurrently with itself. If an OID inconsistency is introduced `gpfilespace --movetempfilespace` or `gpfilespace --movetransfilespace` can be used to revert to the default filespace.
17780	DDL/DML: Partitioning	`ALTER TABLE ADD PARTITION` inheritance issue When performing an `ALTER TABLE ADD PARTITION` operation, the resulting parts may not correctly inherit the storage properties of the parent table in cases such as adding a default partition or more complex subpartitioning. This issue can be avoided by explicitly dictating the storage properties during the `ADD PARTITION` invocation. For leaf partitions that are already afflicted, the issue can be rectified through use of `EXCHANGE PARTITION`.
17795	Management Scripts Suite	Under some circumstances, `gppkg` on SUSE is unable to correctly interpret error messages returned by rpm. On SUSE, `gppkg` is unable to operate correctly under circumstances that require a non-trivial interpretation of underlying rpm commands. This includes scenarios that result from overlapping packages, partial installs, and partial uninstalls.
17604	Security	A Red Hat Enterprise Linux (RHEL) 6.x security configuration file limits the number of processes that can run on gpadmin. RHEL 6.x contains a security file (/etc/security/limits.d/90-nproc.conf) that limits available processes running on gpadmin to 1064. **Workaround**: Remove this file or increase the processes to 131072.
17415	Installer	When you run `gppkg -q -info<some gppkg>`, the system shows the GPDBversion as main build dev.
17334	Management Scripts Suite	You may see warning messages that interfere with the operation of management scripts when logging in. Greenplum recommends that you edit the /etc/motd file and add the warning message to it. This will send the messages to are redirected to `stdout` and not `stderr`. You must encode these warning messages in UTF-8 format.
17221	Resource Management	Resource queue deadlocks may be encountered if a cursor is associated with a query invoking a function within another function.
17113	Management Scripts Suite	Filespaces are inconsistent when the Greenplum database is down. Filespaces become inconsistent in case of a network failure. Greenplum recommends that processes such as moving a filespace be done in an environment with an uninterrupted power supply.
17189	Loaders: gpfdist	`gpfdist` shows the error “Address already in use” after successfully binding to socket IPv6. Greenplum supports IPv4 and IPv6. However, `gpfdist` fails to bind to socket IPv4, and shows the message “Address already in use”, but binds successfully to socket IPv6.

Table 2 All Known Issues in 4.2.7.3

Issue	Category	Description

16519	Backup and Restore	Limited data restore functionality and/or restore performance issues can occur when restoring tables from a full database backup where the default backup directory was not used.
		In order to restore from backup files not located in the default directory you can use the `-R` to point to another host and directory. This is not possible however, if you want to point to a different directory on the same host (NFS for example).
		Workaround: Define a symbolic link from the default dump directory to the directory used for backup, as shown in the following example:
		1. Perform a full Database Backup to a specific NFS directory:
		`$ gpcrondump -x <db_name> -z -u /backup/DCA-93 -a`
		2. Create a file listing the segment servers:
		`$ vi /home/gpadmin/segments`
		`sdw1`
		`sdw2`
		`sdw3`
		`….`
		3. Remove the relevant date folder from db_dumps directories on the master and segments:
		`$ rm -r /data/master/gpseg-1/db_dumps/20120619`
		`$ gpssh -f segments 'rm -r /data1/primary/gpseg*/db_dumps/20120619'`
		`$ gpssh -f segments 'rm -r /data2/primary/gpseg*/db_dumps/20120619'`
		4. Create a symbolic link between the master and segment directories and the directory to which you backed up in step 1. Only the master and sdw1 was shown here, write a script for the remaining segments:
		`$ ln -s /backup/DCA-93/db_dumps/20120619 /data/master/gpseg-1/db_dumps/20120619`
		`$ gpssh -h sdw1 'ln -s /backup/DCA-93/db_dumps/20120619 /data1/primary/gpseg0/db_dumps/20120619'`
		`$ gpssh -h sdw1 'ln -s /backup/DCA-93/db_dumps/20120619 /data1/primary/gpseg1/db_dumps/20120619'`
		`$ gpssh -h sdw1 'ln -s /backup/DCA-93/db_dumps/20120619 /data1/primary/gpseg2/db_dumps/20120619'`
		`$ gpssh -h sdw1 'ln -s /backup/DCA-93/db_dumps/20120619 /data2/primary/gpseg3/db_dumps/20120619'`
		`$ gpssh -h sdw1 'ln -s /backup/DCA-93/db_dumps/20120619 /data2/primary/gpseg4/db_dumps/20120619'`
		`$ gpssh -h sdw1 'ln -s /backup/DCA-93/db_dumps/20120619 /data2/primary/gpseg5/db_dumps/20120619'`
		5. Restore from backup files:
		`$ gpdbrestore -t 20120619061835 -T <schema.table> -a`
		6. Remove the symbolic links
		`$ rrm -r /data/master/gpseg-1/db_dumps/20120619`
		`$ gpssh -f segments 'rm -r /data1/primary/gpseg*/db_dumps/20120619'`
		`$ gpssh -f segments 'rm -r /data2/primary/gpseg*/db_dumps/20120619'`

Table 2 All Known Issues in 4.2.7.3

Issue	Category	Description
16064	Backup and Restore	Restoring a compressed dump with the `--ddboost` option displays incorrect dump parameter information. When using `gpdbrestore --ddboost` to restore a compressed dump, the restore parameters incorrectly show "`Restore compressed dump = Off`". This error occurs even if `gpdbrestore` passes the `--gp-c` option to use gunzip for in-line de-compression.
15899	Backup and Restore	When running `gpdbrestore` with the list (`-L`) option, external tables do not appear; this has no functional impact on the restore job.

Upgrading to Greenplum Database 4.2.x.x

The upgrade path supported for this release is Greenplum Database 4.1.x.x to Greenplum Database 4.2.x.x. The minimum recommended upgrade path for this release is from Greenplum Database version 4.1.1.5. If you have an earlier major version of the database, you must first upgrade to version 4.1.x.

For detailed upgrade procedures and information, see the following sections:

- Upgrading from 4.2.x.x to 4.2.7.3
- Upgrading from 4.1.x.x to 4.2.x.x
- For Users Running Greenplum Database 4.0.x.x
- For Users Running Greenplum Database 3.3.x.x
- Troubleshooting a Failed Upgrade

Upgrading from 4.2.x.x to 4.2.7.3

An upgrade from 4.2.x.x to 4.2.7.3 involves stopping Greenplum Database, updating the Greenplum Database software binaries, and restarting Greenplum Database.

1. Log in to your Greenplum Database master host as the Greenplum administrative user:

   ```
   $ su - gpadmin
   ```

2. Perform a smart shutdown of your current Greenplum Database 4.2.x.x system (there can be no active connections to the database):

   ```
   $ gpstop
   ```

3. Run the installer for 4.2.7.3 on the Greenplum Database master host. When prompted, choose an installation location in the same base directory as your current installation. For example:

   ```
   /usr/local/greenplum-db-4.2.2.x
   ```

4. Edit the environment of the Greenplum Database superuser (gpadmin) and make sure you are sourcing the `greenplum_path.sh` file for the new installation. For example change the following line in `.bashrc` or your chosen profile file:

   ```
   source /usr/local/greenplum-db-4.2.x.x/greenplum_path.sh
   ```

   to:

   ```
   source /usr/local/greenplum-db-4.2.7.3/greenplum_path.sh
   ```

   Or if you are sourcing a symbolic link (`/usr/local/greenplum-db`) in your profile files, update the link to point to the newly installed version. For example:

   ```
   $ rm /usr/local/greenplum-db
   $ ln -s /usr/local/greenplum-db-4.2.7.3
     /usr/local/greenplum-db
   ```

5. Source the environment file you just edited. For example:

   ```
   $ source ~/.bashrc
   ```

6. Run the `gpseginstall` utility to install the 4.2.7.3 binaries on all the segment hosts specified in the *`hostfile`*. For example:

   ```
   $ gpseginstall -f hostfile
   ```

7. After all segment hosts have been upgraded, you can log in as the gpadmin user and restart your Greenplum Database system:

   ```
   $ su - gpadmin
   $ gpstart
   ```

8. If you are utilizing Data Domain Boost, you have to re-enter your DD Boost credentials after upgrading from Greenplum Database 4.2.1 to 4.2.7.3 as follows:

   ```
   gpcrondump --ddboost-host ddboost_hostname --ddboost-user
   ddboost_user
   ```

   Note that if you do not reenter your login credentials after an upgrade, your backup will never start because the Greenplum Database cannot connect to the Data Domain system. You will receive an error advising you to check your login credentials.

Upgrading from 4.1.x.x to 4.2.x.x

This section describes how you can upgrade from Greenplum Database 4.1.x.x or later to Greenplum Database 4.2.x.x. For users running versions prior to 4.1.x.x of Greenplum Database, see the following:

- For Users Running Greenplum Database 4.0.x.x
- For Users Running Greenplum Database 3.3.x.x

Planning Your Upgrade

Before you begin your upgrade, make sure the master and all segments (data directories and filespace) have at least 2GB of free space.

Prior to upgrading your database, Greenplum recommends that you run a pre-upgrade check to verify your database is healthy.

You can perform a pre-upgrade check by executing the `gpmigrator` (`_mirror`) utility with the `--check-only` option.

For example:

```
source $new_gphome/greenplum_path.sh;
gpmigrator_mirror --check-only $old_gphome $new_gphome
```

Note: Performing a pre-upgrade check of your database with the `gpmigrator` (`_mirror`) utility should done during a database maintenance period. When the utility checks the database catalog, users cannot access the database.

Some of the rules for partitioned tables are different in 4.2 than in previous releases. `gpmigrator` detects partitioned tables that violate these new rules and aborts the upgrade. In most cases, `gpmigrator` will create a repair script you can run to bring your 4.1 Greenplum Database into line with the new rules in 4.2. See Upgrading Partitioned Tables with Constraints for more details.

Upgrading Partitioned Tables with Constraints

Partition tables with `CHECK`, `PRIMARY KEY`, or `UNIQUE` constraints must be updated prior to upgrading:

- Regular `CHECK`, `PRIMARY KEY`, or `UNIQUE` constraints added by database users usually appear on every sub-table of the partitioned table. Their names may need to be adjusted. If necessary, `gpmigrator` creates a repair script to do this.
- `PRIMARY KEY` and `UNIQUE` constraints on partitioned tables that do not include all the columns of the partition key need to be removed. If needed, `gpmigrator` creates a repair script to do this. Note that the unique index underlying the constraint remains and provides the same protection against duplicate keys as did the constraint. As was previously the case, it is possible for different parts of the partitioned table to contain the same key value.
- If the added constraints appear on some but not all of the sub-tables that make up a partitioned table, they cannot be updated automatically. In this case, you need to either drop the irregular constraints or add the missing constraints. Other causes of irregularity may exist but are rare. Greenplum recommends that you contact support if you encounter any issues with partitioned tables that cannot be resolved automatically with `gpmigrator`.

Upgrade Procedure

This section divides the upgrade into the following phases: pre-upgrade preparation, software installation, upgrade execution, and post-upgrade tasks.

We have also provided you with an Upgrade Checklist that summarizes this procedure.

Important: Carefully evaluate each section and perform all required and conditional steps. Failing to perform any of these steps can result in an aborted upgrade, placing your system in an unusable or even unrecoverable state.

Pre-Upgrade Preparation (on your 4.1.x system)

Perform these steps on your current 4.1.x Greenplum Database system. This procedure is performed from your Greenplum master host and should be executed by the Greenplum superuser (`gpadmin`).

1. Log in to the Greenplum Database master as the `gpadmin` user:

   ```
   $ su - gpadmin
   ```

2. (*optional*) Vacuum all databases prior to upgrade. For example:

   ```
   $ vacuumdb database_name
   ```

3. (*optional*) Clean out old server log files from your master and segment data directories. For example, to remove log files from 2011 from your segment hosts:

   ```
   $ gpssh -f seg_host_file -e 'rm
   /gpdata/*/gp*/pg_log/gpdb-2011-*.csv'
   ```

Note: Running Vacuum and cleaning out old logs files is not required, but it will reduce the size of Greenplum Database files to be backed up and migrated.

4. Run `gpstate` to check for failed segments.

   ```
   $ gpstate
   ```

5. If you have failed segments, you must recover them using `gprecoverseg` before you can upgrade.

   ```
   $ gprecoverseg
   ```

   Note: It might be necessary to restart the database if the preferred role does not match the current role; for example, if a primary segment is acting as a mirror segment or a mirror segment is acting as a primary segment.

6. Copy or preserve any additional folders or files (such as backup folders) that you have added in the Greenplum data directories or `$GPHOME` directory. Only files or folders strictly related to Greenplum Database operations are preserved by the migration utility.

Install the Greenplum Software Binaries

1. Download or copy the installer file to the Greenplum Database master host.

2. Unzip the installer file. For example:

   ```
   # unzip greenplum-db-4.2.x.x-PLATFORM.zip
   ```

3. Launch the installer using `bash`. For example:

   ```
   # /bin/bash greenplum-db-4.2.x.x-PLATFORM.bin
   ```

4. The installer will prompt you to accept the Greenplum Database license agreement. Type `yes` to accept the license agreement.

5. The installer will prompt you to provide an installation path. Press `ENTER` to accept the default install path (for example: `/usr/local/greenplum-db-4.2.x.x`), or enter an absolute path to an install location. You must have write permissions to the location you specify.

6. The installer will install the Greenplum software and create a `greenplum-db` symbolic link one directory level above your version-specific Greenplum installation directory. The symbolic link is used to facilitate patch maintenance and upgrades between versions. The installed location is referred to as `$GPHOME`.

7. Source the path file from your new 4.2.x.x installation. For example:

   ```
   $ source /usr/local/greenplum-db-4.2.x.x/greenplum_path.sh
   ```

8. Run the `gpseginstall` utility to install the 4.2.3.x binaries on all the segment hosts specified in the *`hostfile`*. For example:

   ```
   $ gpseginstall -f hostfile
   ```

Upgrade Execution

During upgrade, all client connections to the master will be locked out. Inform all database users of the upgrade and lockout time frame. From this point onward, users should not be allowed on the system until the upgrade is complete.

9. Source the path file from your old 4.1.x.x installation. For example:

   ```
   $ source /usr/local/greenplum-db-4.1.1.5/greenplum_path.sh
   ```

10. (*optional but strongly recommended*) Back up all databases in your Greenplum Database system using `gpcrondump` (or zfs snapshots on Solaris systems). See the *Greenplum Database Administrator Guide* for more information on how to do backups using `gpcrondump`. Make sure to secure your backup files in a location outside of your Greenplum data directories.

11. If your system has a standby master host configured, remove the standby master from your system configuration. For example:

    ```
    $ gpinitstandby -r
    ```

12. Perform a clean shutdown of your current Greenplum system. For example:

    ```
    $ gpstop
    ```

13. Source the path file from your new 4.2.x.x installation. For example:

    ```
    $ source /usr/home/greenplum-db-4.2.x.x/greenplum_path.sh
    ```

14. Update your environment so it is sourcing your new 4.2.x.x installation.

    a. For example, update the `greenplum-db` symbolic link on the master and standby master to point to the new 4.2.x.x installation directory. For example (as root):

    ```
    # rm -rf /usr/local/greenplum-db
    # ln -s /usr/local/greenplum-db-4.2.x.x
    /usr/local/greenplum-db
    # chown -R gpadmin /usr/local/greenplum-db
    ```

    b. Using `gpssh`, also update the `greenplum-db` symbolic link on all of your segment hosts. For example (as root):

    ```
    # gpssh -f segment_hosts_file
    => rm -rf /usr/local/greenplum-db
    => ln -s /usr/local/greenplum-db-4.2.x.x
    /usr/local/greenplum-db
    => chown -R gpadmin /usr/local/greenplum-db
    => exit
    ```

15. (*optional but recommended*) Prior to running the migration, perform a pre-upgrade check to verify that your database is healthy by executing the 4.2.x.x version of the `gpmigrator` utility with the `--check-only` option. For example:

    ```
    # gpmigrator_mirror --check-only
    /usr/local/greenplum-db-4.1.1.5
    /usr/local/greenplum-db-4.2.x.x
    ```

16. As `gpadmin`, run the 4.2.x.x version of the migration utility specifying your old and new `GPHOME` locations. If your system has mirrors, use `gpmigrator_mirror`. If your system does not have mirrors, use `gpmigrator`. For example on a system with mirrors:

    ```
    $ su - gpadmin
    $ gpmigrator_mirror /usr/local/greenplum-db-4.1.1.5
    /usr/local/greenplum-db-4.2.x.x
    ```

Note: If the migration does not complete successfully, contact Customer Support (see "Troubleshooting a Failed Upgrade" on page 21).

17. The migration can take a while to complete. After the migration utility has completed successfully, the Greenplum Database 4.2.x.x system will be running and accepting connections.

    Note: After the migration utility has completed, the resynchronization of the mirror segments with the primary segments continues. Even though the system is running, the mirrors are not active until the resynchronization is complete.

Post-Upgrade (on your 4.2.x.x system)

18. If your system had a standby master host configured, reinitialize your standby master using `gpinitstandby`:

    ```
    $ gpinitstandby -s standby_hostname
    ```

19. If your system uses external tables with `gpfdist`, stop all `gpfdist` processes on your ETL servers and reinstall `gpfdist` using the compatible Greenplum Database 4.2.x.x Load Tools package. Application Packages are available from the EMC Download Center.

20. Rebuild any custom modules against your 4.2.x.x installation (for example, any shared library files for user-defined functions in `$GPHOME/lib`).

21. Greenplum Database 4.2.x.x introduced the `gppkg` utility to install Greenplum Database extensions. If you were previously using any PostgreSQL extensions such as pgcrypto, PL/R, PL/Java, PL/Perl, and PostGIS, download the corresponding packages from the EMC Download Center, and install using this new utility. See the *Greenplum Database Administrator Guide 4.2* or later for usage details.

22. If you want to utilize the Greenplum Command Center management tool, install the latest Command Center Console and update your environment variable to point to the latest Command Center binaries (source the `gpperfmon_path.sh` file from your new installation).

    Note that the Greenplum Command Center management tool replaces Greenplum Performance Monitor.

    Command Center Console packages are available from the EMC Download Center or Pivotal Network.

23. Inform all database users of the completed upgrade. Tell users to update their environment to source the Greenplum Database 4.2.x.x installation (if necessary).

Upgrade Checklist

This checklist provides a quick overview of all the steps required for an upgrade from 4.1.x.x to 4.2.x.x. Detailed upgrade instructions are provided in the Upgrade Procedure section.

Pre-Upgrade Preparation (on your current system)	
* 4.1.x.x system is up and available	
☐	Log in to your master host as the `gpadmin` user (your Greenplum superuser).
☐	(Optional) Run `VACUUM` on all databases,
☐	(Optional) Remove old server log files from `pg_log` in your master and segment data directories.
☐	Check for and recover any failed segments (`gpstate`, `gprecoverseg`).
☐	Copy or preserve any additional folders or files (such as backup folders).
☐	Install the Greenplum Database 4.2.x.x binaries on all Greenplum hosts.
☐	Inform all database users of the upgrade and lockout time frame.
Upgrade Execution	
* The system will be locked down to all user activity during the upgrade process	
☐	Backup your current databases.
☐	Remove the standby master (`gpinitstandby -r`).
☐	Do a clean shutdown of your current system (`gpstop`).
☐	Update your environment to source the new Greenplum Database 4.2.x.x installation.
☐	Run the upgrade utility (`gpmigrator_mirror` if you have mirrors, `gpmigrator` if you do not).
☐	After the upgrade process finishes successfully, your 4.2.x.x system will be up and running.
Post-Upgrade (on your 4.2.x.x system)	
* The 4.2.x.x system is up	
☐	Reinitialize your standby master host (`gpinitstandby`).

☐	Upgrade `gpfdist` on all of your ETL hosts.
☐	Rebuild any custom modules against your 4.2.x.x installation.
☐	Download and install any Greenplum Database extensions.
☐	(Optional) Install the latest Command Center Console and update your environment to point to the latest Command Center binaries.
☐	Inform all database users of the completed upgrade.

For Users Running Greenplum Database 4.0.x.x

Users on a release prior to 4.1.x.x cannot upgrade directly to 4.2.x.x.

- Upgrade from your current release to 4.1.x.x (follow the upgrade instructions in the latest Greenplum Database 4.1.x.x release notes available on Support Zone).
- Follow the upgrade instructions in these release notes for Upgrading from 4.1.x.x to 4.2.x.x.

For Users Running Greenplum Database 3.3.x.x

Users on a release prior to 4.0.x.x cannot upgrade directly to 4.1.x.

- Upgrade from your current release to 4.0.x.x (follow the upgrade instructions in the latest Greenplum Database 4.0.x.x release notes available on Support Zone).
- Upgrade the 4.0.x.x release to 4.1.x.x (follow the upgrade instructions in the latest Greenplum Database 4.1.x.x release notes available on Support Zone).
- Follow the upgrade instructions in these release notes for Upgrading from 4.1.x.x to 4.2.x.x.

Troubleshooting a Failed Upgrade

If you experience issues during the migration process, go to the Support page at Support Zone or contact Greenplum customer support at one of the following numbers:

United States: 800-782-4362 (1-800-SVC-4EMC)

Canada: 800-543-4782

Worldwide: +1-508-497-7901

Be prepared to provide the following information:

- A completed Upgrade Procedure.
- Log output from `gpmigrator` and `gpcheckcat` (located in `~/gpAdminLogs`)

Greenplum Database Tools Compatibility

Client Tools

Greenplum releases a number of client tool packages on various platforms that can be used to connect to Greenplum Database and the Greenplum Command Center management tool. The following table describes the compatibility of these packages with this Greenplum Database release.

Tool packages are available from the EMC Download Center.

Table 3 Greenplum Database Tools Compatibility

Client Package	Description of Contents	Client Version	Server Versions
Greenplum Clients	Greenplum Database Command-Line Interface (psql) Greenplum MapReduce (gpmapreduce)[1]	4.2.7	4.2.7
Greenplum Connectivity	Standard PostgreSQL Database Drivers (ODBC, JDBC) PostgreSQL Client C API (libpq)	4.2.7	4.2.7
Greenplum Loaders	Greenplum Database Parallel Data Loading Tools (gpfdist, gpload)	4.2.7	4.2.7
Greenplum Command Center	Greenplum Database management tool.	1.2.0.1	4.2.7

1. gpmapreduce is not available on Windows.

The Greenplum Database Client Tools, Load Tools, and Connectivity Tools are supported on these platforms:

- AIX 5.3L (32-bit)
- AIX 5.3L and AIX 6.1 (64-bit)
- Apple OSX on Intel processors (32-bit)
- HP-UX 11i v3 (B.11.31) Intel Itanium (Client and Load Tools only)
- Red Hat Enterprise Linux i386 (RHEL 5)
- Red Hat Enterprise Linux x86_64 (RHEL 4)

- Red Hat Enterprise Linux x86_64 (RHEL 5 and RHEL 6)
- Solaris 10 i386
- Solaris 10 SPARC32
- Solaris 10 SPARC64
- Solaris 10 x86_64
- Solaris 9 SPARC32
- SUSE Linux Enterprise Server x86_64 (SLES 10 and SLES 11)
- Windows 7 (32-bit and 64-bit)
- Windows Server 2003 R2 (32-bit and 64-bit)
- Windows Server 2008 R2 (64-bit)
- Windows XP (32-bit and 64-bit)

Greenplum GPText

GPText enables processing mass quantities of raw text data (such as social media feeds or e-mail databases) into mission-critical information that guides business and project decisions. GPText joins the Greenplum Database massively parallel-processing database server with Apache Solr enterprise search.

GPText requires Greenplum Database. See the GPText release notes for the required version of Greenplum Database.

Greenplum Database Extensions Compatibility

Greenplum Database delivers an agile, extensible platform for in-database analytics, leveraging the system's massively parallel architecture. With Release 4.2.x.x, Greenplum enables turn-key in-database analytics via Greenplum Extensions.

You can download Greenplum extensions packages from the EMC Download Center, and install them using the Greenplum Packager Manager (`gppkg`). See the *Greenplum Database Administrator Guide 4.2* or later for details.

Note that Greenplum Package Manager installation files for extension packages may release outside of standard Database release cycles. Therefore, for the latest install and configuration information regarding any supported database package/extension, go to the `Support` site and download `Primus Article 288189` from our knowledge base (Requires a valid login to the EMC Support site).

The following table provides information about the compatibility of the Greenplum Database Extensions and their components with this Greenplum Database release.

Note that the PL/Python database extension is already included with the standard Greenplum database distribution.

Table 4 Greenplum Database Extensions Compatibility

Greenplum Database Extension	Extension Components	
	Name	**Version**
PostGIS 2.0 for Greenplum Database 4.2.x.x	PostGIS	2.0.3
	Proj	4.8.0
	Geos	3.3.8
PostGIS 1.0 for Greenplum Database 4.2.x.x	PostGIS	1.4.2
	Proj	4.7.0
	Geos	3.2.2
PL/Java 1.0 for Greenplum Database 4.2.x.x	PL/Java	Based on 1.4.0
	Java JDK	1.6.0_26 Update 31
PL/R 1.0 for Greenplum Database 4.2.x.x	PL/R	8.3.0.12
	R	2.13.0
PL/Perl 1.2 for Greenplum Database 4.2.x.x	PL/Perl	Based on PostgreSQL 9.1
	Perl	5.12.4 on RHEL 6.x 5.5.8 on RHEL 5.x, SUSE 10
PL/Perl 1.1 for Greenplum Database 4.2.x.x	PL/Perl	Based on PostgreSQL 9.1
	Perl	5.12.4 on RHEL 5.x, SUSE 10
PL/Perl 1.0 for Greenplum Database 4.2.x.x	PL/Perl	Based on PostgreSQL 9.1
	Perl	5.12.4 on RHEL 5.x, SUSE 10
Pgcrypto 1.0 for Greenplum Database 4.2.x.x	Pgcrypto	Based on PostgreSQL 8.3
Greenplum Hadoop File System	gphdfs	1.1
	gphdfs	1.2
	gphdfs	1.3
	gphdfs	1.4
MADlib 1.4 for Greenplum Database 4.2.x.x	MADlib	Based on MADlib version 0.7.0

Hadoop Distribution Compatibility

Use the gppkg utility to install the gNet package containing the jar file for the extensions, the libraries, and the documentation for the gphdfs extensions. To install the correct distribution, refer to the following Hadoop extensions compatibility matrix:

Table 5 Hadoop Extensions Compatibility

Hadoop Distribution	Version
Pivotal HD	Pivotal HD 1.0[1]
Greenplum HD	Greenplum HD 1.1
	Greenplum HD 1.2
Cloudera	cdh3u2
	cdh3u4
	CDH4.1 with MRv1
Greenplum MR	Greenplum MR 1.0
	Greenplum MR 1.2

1. A distribution of Hadoop 2.0

Greenplum Database 4.2 Documentation

For the latest Greenplum Database documentation go to Pivotal Documentation. Greenplum documentation is provided in PDF format.

Table 6 Greenplum Database Documentation

Title	Revision
Greenplum Database 4.2.7.3 Release Notes	A01
Greenplum Database 4.2 Installation Guide	A11
Greenplum Database 4.2 Database Administrator Guide	A08
Greenplum Database 4.2 System Administrator Guide	A12
Greenplum Database 4.2 Reference Guide	A12
Greenplum Database 4.2 Utility Guide	A12
Greenplum Database 4.2 Client Tools for UNIX	A05
Greenplum Database 4.2 Client Tools for Windows	A05
Greenplum Database 4.2 Connectivity Tools for UNIX	A04
Greenplum Database 4.2 Connectivity Tools for Windows	A03
Greenplum Database 4.2 Load Tools for UNIX	A05
Greenplum Database 4.2 Load Tools for Windows	A05
Greenplum Command Center 1.2 Administrator Guide	A01

Copyright © 2014 Pivotal Software, Inc. All rights reserved.

Pivotal Software, Inc. believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

THE INFORMATION IN THIS PUBLICATION IS PROVIDED "AS IS." PIVOTAL SOFTWARE, INC. ("Pivotal") MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND WITH RESPECT TO THE INFORMATION IN THIS PUBLICATION, AND SPECIFICALLY DISCLAIMS IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Use, copying, and distribution of any Pivotal software described in this publication requires an applicable software license.

All trademarks used herein are the property of Pivotal or their respective owners.